WO03007596

Publication Title:

ELECTRONIC PROGRAM GUIDE FOR PROCESSING CONTENT-RELATED INFORMATION CONFIGURED USING A REFERENCE INFORMATION MODEL

Abstract:

Abstract of WO 03007596

(A1) Translate this text Electronic program guide (EPG) processing techniques are disclosed which involve processing content-related information in the form of documents generated using a reference information model (RIM). The documents may be configured in an extensible mark-up language (XML) or other standard format. At least a portion of the content-related information is configured for consistency with corresponding portions of the RIM, the portion of the content-related information so configured thereby being suitable for processing by different EPG applications. The RIM preferably comprises multiple classes of information, and specifies properties of the classes utilizing attributes, relationships and states. For example, instances of the classes may be configured as objects in an object-oriented programming format, and one or more of the objects may contain structures represented as attributes. The RIM may be generated utilizing an iterative process in which progressively more inclusive versions are generated by modifying previous versions to support additional data specifications.

Courtesy of http://v3.espacenet.com

This Patent PDF Generated by Patent Fetcher(R), a service of Stroke of Color, Inc.

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
23 January 2003 (23.01.2003)

PCT

(10) International Publication Number
WO 03/007596 A1

(51) International Patent Classification[7]: H04N 5/445, G06F 17/30

(21) International Application Number: PCT/IB02/02568

(22) International Filing Date: 24 June 2002 (24.06.2002)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
09/902,185 10 July 2001 (10.07.2001) US

(71) Applicant: KONINKLIJKE PHILIPS ELECTRONICS N.V. [NL/NL]; Groenewoudseweg 1, NL-5621 BA Eindhoven (NL).

(72) Inventors: ALSAFADI, Yasser; Internationaal Octrooibureau B.V., Prof . Holstlaan 6, NL-5656 AA Eindhoven (NL). YASSIN, Amr, F.; Internationaal Octrooibureau B.V., Prof . Holstlaan 6, NL-5656 AA Eindhoven (NL). TIRADO-RAMOS, Alfredo; Internationaal Octrooibureau B.V., Prof . Holstlaan 6, NL-5656 AA Eindhoven (NL).

(74) Agent: GROENENDAAL, Antonius, W., M.; Internationaal Octrooibureau B.V., Prof. Holstlaan 6, NL-5656 AA Eindhoven (NL).

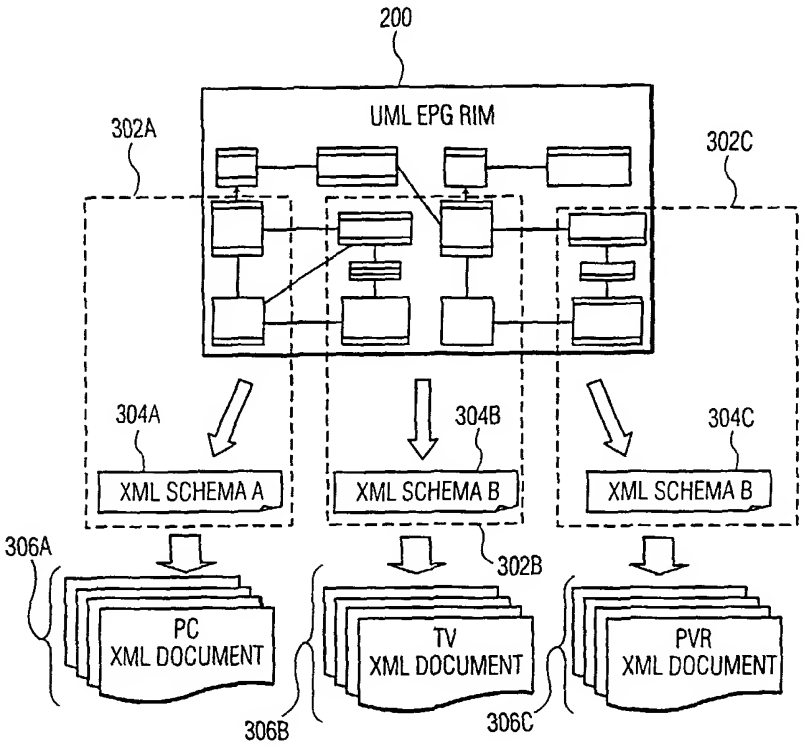
(81) Designated States *(national)*: CN, JP, KR.

(84) Designated States *(regional)*: European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR).

Published:
— *with international search report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: ELECTRONIC PROGRAM GUIDE FOR PROCESSING CONTENT-RELATED INFORMATION CONFIGURED USING A REFERENCE INFORMATION MODEL



(57) Abstract: Electronic program guide (EPG) processing techniques are disclosed which involve processing content-related information in the form of documents generated using a reference information model (RIM). The documents may be configured in an extensible mark-up language (XML) or other standard format. At least a portion of the content-related information is configured for consistency with corresponding portions of the RIM, the portion of the content-related information so configured thereby being suitable for processing by different EPG applications. The RIM preferably comprises multiple classes of information, and specifies properties of the classes utilizing attributes, relationships and states. For example, instances of the classes may be configured as objects in an object-oriented programming format, and one or more of the objects may contain structures represented as attributes. The RIM may be generated utilizing an iterative process in which progressively more inclusive versions are generated by modifying previous versions to support additional data specifications.

WO 03/007596 A1

Electronic program guide for processing content-related information configured using a reference information model

The present invention relates generally to electronic program guides (EPGs) utilized to facilitate access to content, and more particularly to techniques for configuring EPGs to process content-related information in the form of extensible mark-up language (XML) documents.

EPGs are essential tools for management of the ever-increasing array of content available from information sources such as broadcast, satellite, cable and the Internet. As is well known, EPGs are generally associated with televisions, set-top boxes, personal video recorders (PVRs) or other similar devices, and can provide users of such devices with detailed information regarding available content, including titles, transmission times, actor lists, ratings, recommendations, etc.

A significant problem with existing EPGs is that there is no mechanism available to provide sufficient compatibility between the widely diverse applications that generate content. As a result, content generated by one application for a device having a particular type of EPG may not be configured in a manner suitable for processing by another device having a different type of EPG. Different devices are therefore generally required to have different EPGs, each specifically configured to operate with the particular content accessible via the corresponding device. This conventional approach requires excessive duplication of EPG resources, and unduly limits user accessibility to content-related information.

It is also well known to utilize standard formats for delivery of content over computer networks such as the Internet. An example of one such standard format is the extensible mark-up language (XML), described in XML 1.0 (Second Edition), World Wide Web Consortium (W3C) Recommendation, October 2000, www. w3.org/TR/REC-xml, which is incorporated by reference herein. However, XML and other similar standard formats often fail to capture adequately the semantics used in content generation for different applications. For example, there is considerable heterogeneity in the semantics of EPG field names, reflecting a lack of agreement among system developers in this area. Therefore, sets of XML documents from different sources and corresponding EPG-equipped devices will

generally not have the desired interoperability in the absence of a difficult-to-obtain specific prior agreement upon semantic definitions.

As is apparent from the foregoing, a need exists in the art for techniques that can allow different types of content from different sources to be configured in a standardized manner suitable for efficient processing by different EPGs, while avoiding the above-noted problems associated with the conventional approaches.

The present invention meets the above-noted need by providing an electronic program guide (EPG) for processing of extensible mark-up language (XML) documents and other content-related information configured in accordance with a reference information model (RIM). The RIM is designed such that XML documents or other content-related information can be generated therefrom in a consistent manner so as to be suitable for processing by a wide variety of different EPG applications.

In accordance with one aspect of the invention, at least a portion of the content-related information is configured for consistency with corresponding portions of the RIM, the portion of the content-related information so configured thereby being suitable for processing by different EPG applications.

The RIM preferably comprises multiple classes of information, and specifies properties of the classes utilizing attributes, relationships and states. For example, instances of the classes may be configured as objects in an object-oriented programming format, and one or more of the objects may contain structures represented as attributes. In an illustrative embodiment, the RIM includes multiple enumeration elements and multiple class elements, with a given one of the class elements being associated with at least a subset of the enumeration elements and at least a subset of the remaining class elements. By way of example, the given class element for content may be a program class element, and the remaining class elements may include class elements for movie, episode, personnel, cast, credits, station and designated market area.

In accordance with another aspect of the invention, the RIM may be generated utilizing an iterative process in which progressively more inclusive versions are generated by modifying previous versions to support additional data specifications. For example, an initial version of the RIM may be generated using a first set of data specifications, and at least one subsequent version of the RIM may be generated from the initial version using at least a second set of data specifications. The subsequent version of the RIM may be periodically updated in accordance with one or more sets of updated data specifications, as such updated specifications become available.

In accordance with yet another aspect of the invention, the content-related information may be transformed from a first format not compliant with the RIM to a second format compliant with the RIM. The content-related information in the first format may comprise one or more documents for use with an EPG application of a type not based on the RIM, and may be subsequently converted into a desired RIM-compliant format. The transformation may utilize an XML style sheet generated at least in part utilizing the RIM and schema derived from the content-related information in the first format.

Advantageously, the present invention through use of the RIM ensures that content from diverse information sources can be processed in an efficient and effective manner by different EPGs, e.g., multiple EPGs associated with different processing devices of a residence, business or other user location. In addition, it allows XML documents developed for use with particular types of EPGs to be utilizable in an efficient manner on other EPGs. The invention also eases semantic verification of conformance claims between EPG applications from different developers, and can improve the marketability of a given EPG-based processing device. Moreover, the use of the RIM in the manner described can ensure internal consistency across the product lines of a given device manufacturer.

The invention thus facilitates the processing of EPG XML documents and other content-related information so as to deliver consistent results in an efficient manner.

These and other features and advantages of the present invention will become more apparent from the accompanying drawings and the following detailed description.

Fig. 1 is a block diagram of an example information processing system in which the present invention is implemented.

Fig. 2 shows a portion of an example reference information model (RIM) in accordance with an illustrative embodiment of the invention.

Fig. 3 illustrates the manner in which different types of documents may be generated utilizing a RIM such as that shown in Fig. 2.

Fig. 4 is a flow diagram of a process for generating, adjusting and updating a RIM such as that shown in Fig. 2.

Fig. 5 is a flow diagram of a process for transforming a set of documents, utilizing a RIM such as that shown in Fig. 2.

Fig. 6 shows one possible embodiment of a device in which an electronic program guide (EPG) in accordance the invention is implemented.

The present invention will be illustrated herein using an example information processing system and reference information model (RIM) for standardizing content-related information for processing by electronic program guides (EPGs). It should be understood, however, that the invention does not require the use of any particular system, RIM or EPG configuration, and is more generally suitable for use in any application involving an EPG in which it is desirable to provide improved processing of documents or other information in an extensible mark-up language (XML) format or other similar standardized format.

Fig. 1 shows an information processing system 100 that in this illustrative embodiment includes a residence 102 coupled to an external network 104. The residence 102 includes a number of processing devices 110 and corresponding EPGs 112. More particularly, the residence 102 includes a video game console 110-1, a television (TV) 110-2, a personal video recorder (PVR) 110-3, a set-top box 110-4, a music jukebox 110-5 and associated audio system 110-6, and a browser-equipped personal computer (PC) 110-7. Each of the processing devices 110-1, 110-2, . . . 110-7 has associated therewith a corresponding EPG 112-1, 112-2, . . . 112-7, respectively. The processing devices 110 are coupled to a home network 115 as shown.

Although described in conjunction with residence 102 in this embodiment, the invention can be implemented with any arrangement of one or more processing devices, e.g., mobile devices used away from a residence, or devices used within a business location or other enterprise. Moreover, although the EPGs 112 are shown in Fig. 1 as being separate from their corresponding processing devices 110, the EPGs 112 are typically implemented within their corresponding devices.

The external network 104 supplies content from one or more information sources to the processing devices 110 of the residence 102. The network 104 may represent a combination of multiple conventional networks, such as telephone, broadcast, satellite, cable, Internet and other networks, each being coupled to the residence 102 via a given channel in a set of communication channels 115.

By way of example, a single information source in the form of a web server 120 is shown in the figure, although numerous other information sources could be used in place of or in conjunction with the web server 120. The web server 120 provides content services to one or more of the processing devices 110 of the residence 102. Associated with the web server 120 is an EPG 122. The web server 120 and its associated EPG 122 may be

configured to deliver content to the processing devices 110 in the form of XML documents transmitted over an Internet portion of the network 104, using well-known techniques such as Internet protocol (IP) and transmission control protocol (TCP).

The EPGs 112 in this illustrative embodiment preferably include conventional XML parsers that permit extraction of content-related information from documents supplied by the web server 120 or other information source.

An XML parser may be viewed as a software library used to facilitate XML document manipulations. Examples of conventional XML parsers include the Xerecs-J and Xerecs-C parsers, and the XP parser. Standard application programming interfaces (APIs) are used to provide predefined interfaces for one or more of these parsers. These APIs include DOM 1.0, described in Document Object Model (DOM) Level 1 Specification, Version 1.0, W3C Recommendation, October 1998, www.w3.org/TR/1998/REC-DOM-Level-1-19981001, which is incorporated by reference herein, and SAX, described in SAX 2.0, "The Simple API for XML," www.megginson.com/SAX/sax.html, which is incorporated by reference herein. The above-noted Xerecs-J and Xerecs-C parsers implement both the DOM and SAX APIs, while the XP parser implements only the SAX API.

The term "document" as used in this context is intended to include not only XML documents, but any other file or arrangement of information that may be transmitted from an information source of the system 100 for utilization by one or more of the processing devices 110 and/or an associated EPG 112.

The term "content-related information" as used herein is intended to be construed broadly so as to include content itself, as well as information characterizing content, and may be in the form of one or more documents.

It should be noted that the particular arrangement and configuration of elements shown in system 100 of Fig. 1 are by way of example only. In other embodiments, other types of servers, networks and processing devices may be used. Those skilled in the art will recognize that the EPG techniques of the present invention do not require any particular arrangement or configuration of such system elements.

In accordance with one aspect of the present invention, one or more of the EPGs 112 associated with residence 102 are configured to process XML documents generated using an EPG reference information model (RIM). Advantageously, the use of the RIM ensures that content from diverse information sources can be processed in an efficient and effective manner by different EPGs, e.g., the EPGs 112 of residence 102 in Fig. 1.

Fig. 2 shows an example of a RIM 200 in accordance with an illustrative embodiment of the invention. As will be described in detail below, the RIM 200 may be used to generate particular types of documents, e.g., XML documents, suitable for use with one or more of the processing devices 110 of the residence 102.

The RIM 200 includes interrelated elements arranged as shown, and is configured at least in part to model content-related information in a format similar to that described in Tribune Media Services, "Data specifications: TV schedules, U.S." Version 2.0, Tribune Media Services, January 29, 1999, and program listings from TV Data Technologies, www.tvdata.com, both of which are incorporated by reference herein.

Enumeration is used to describe constants within the modeled information. More particularly, the RIM 200 uses enumeration to describe constant elements 202-1, 202-2, . . . 202-15. The RIM 200 further includes class elements 204, 206, 208, 210, 212, 214, 216 and 218 for information classes denoted as program, movie, episode, personnel, cast, credits, station and designated market area, respectively. Additional details regarding each of the elements of the example RIM 200 can be found in the attached Appendix 1.

It can be seen from FIG. 2 that the program class 204 has directly or indirectly associated therewith the constant elements 202-1 through 202-15. Moreover, the program class 204 is directly or indirectly associated with each of the other classes 206, 208, 210, 212, 214, 216 and 218 as shown.

The RIM 200 in the illustrative embodiment of FIG. 2 is an object-oriented model implemented using the well-known Unified Modeling Language (UML). UML is a preferred modeling language in that it provides a modeling notation having well-defined semantics that can be interpreted unambiguously, but other types of modeling languages may be used to generate a RIM in accordance with the invention. It should also be noted that enhancements to the modeling approach of the illustrative embodiment can be enhanced by the use of Object Constraint Language (OCL)-compliant UML tools, as described at www.omg.org.

The RIM 200 may also make use of conventional information model techniques such as those described in "Message Development Framework," HL7 Modeling and Methodology Committee, Version 3.2, April 1999.

The RIM 200 In the illustrative embodiment comprises a structured specification of the information requirements associated with one or more types of content. The RIM 200 expresses the classes of information required, and the properties of those classes including attributes, relationships, and states.

More particularly, the RIM 200 may be viewed as a structured specification of information requirements regarding exchange of content-related information. The RIM 200 provides a consistent view of the information being transmitted, as well as relationships to other information, in accordance with a specified semantic and syntactic consensus. This ensures that the content-related information exchanged is consistent and can be used by different EPG applications. The term "EPG application" as used herein is intended to include any particular instance of an EPG as implemented in conjunction with a given processing device, e.g., a given one of the EPGs 112 or 122 as illustrated in FIG. 1, as well as portions or combinations of these and other EPGs.

Advantageously, the RIM 200 includes well-defined classes, which represent content-related information suitable for processing by an EPG, and specifies relationships among such classes. In a typical implementation, as is apparent from the foregoing example of FIG. 2, individual instances of these classes exist as objects, and contain structures represented as attributes.

Fig. 3 illustrates the manner in which different types of XML documents may be generated utilizing the RIM 200 of FIG. 2. In the illustrated process, the RIM 200 is utilized to generate XML documents for three different processing devices of the FIG. 1 system, namely, a PC, TV and PVR, shown as respective elements 110-7, 110-2 and 110-3 in FIG. 1. More particularly, operations 302A, 302B, and 302C utilize respective first, second and third portions of the RIM 200 shown as falling within corresponding dashed boxes. The operations 302A, 302B and 302C generate different XML schema 304A, 304B and 304C, respectively. The XML schema 304A, 304B and 304C, also denoted as XML Schema A, XML Schema B and XML Schema C in the figure, are then used to generate sets of XML documents 306A, 306B and 306C for the PC, TV and PVR, respectively. The schema 304A, 304B and 304C may represent, e.g., normative portions of EPG configurations for particular processing devices as determined by standards bodies for presentation of content on those devices.

Advantageously, this process provides the desired consistency between XML documents or other content-related information generated for different EPG applications associated with different processing devices. The invention in other embodiments can provide similar advantages for different EPG applications running on a single processing device.

The Fig. 3 process can be implemented at least in part using otherwise conventional XML document generation tools such as those commercially available from Rational, www.rational.com, XML Authority, www.extensibility.com, and others.

An example XML schema generated from a portion of the RIM 200 using the XML Authority 1.2 toolkit is as follows:

```
<?xml version ="1.0"?>
<!--Conforms to w3c http://www.w3.org/1999/XMLSchema-->
<schema xmlns = "http://www.w3.org/1999/XMLSchema">
    <element name = "EPG">
        <complexType content = "elementOnly">
            <sequence>
                <element ref = "Program"/>
            </sequence>
        </complexType>
    </element>
    <element name = "Program">
        <complexType content = "elementOnly">
            <sequence>
                <element ref = "AdvisoryDescription"/>
            </sequence>
            <attribute name = "uid" type = "string"/>
            <attribute name = "title" type = "string"/>
            <attribute name = "advisory_description" type = "string"/>
            <attribute name = "air_date" type = "string"/>
            <attribute name = "air_time" type = "string"/>
            <attribute name = "duration" type = "string"/>
        </complexType>
    </element>
    <element name = "AdvisoryDescription">
        <complexType base = "NMTOKEN" content = "textOnly">
            <attribute name = "AdultSituations" type = "string"/>
            <attribute name = "AdolescentesYAdultos" type = "string"/>
            <attribute name = "Adultos" type = "string"/>
```

```
                    <attribute name = "BriefNudity" type = "string"/>
                    <attribute name = "GraphicLanguage" type = "string"/>
                    <attribute name = "GraphicViolence" type = "string"/>
                    <attribute name = "Language" type = "string"/>
                    <attribute name = "MildViolence" type = "string"/>
                    <attribute name = "Nudity" type = "string"/>
                    <attribute name = "PublicoGeneral" type = "string"/>
                    <attribute name = "Rape" type = "string"/>
                    <attribute name = "StrongSexualContent" type = "string"/>
                    <attribute name = "Violence" type = "string"/>
                </complexType>
            </element>
        </schema>
```

As is apparent from this example, complete XML schema can be generated from the RIM 200 or from one or more suitable subsets thereof.

In accordance with another aspect of the invention, an iterative process for generating, adjusting and updating a RIM is provided.

Fig. 4 is a flow diagram of the iterative process. In this process a set of U.S. EPG data specifications 402 is utilized in generation step 404 to generate a U.S. EPG RIM 406. An example of such a U.S. EPG RIM is the RIM 200 previously described in conjunction with FIGS. 2 and 3. The U.S. EPG RIM 406 is then adjusted in an adjustment step 408 utilizing Europe EPG data specifications 410. The result of the adjustment step 408 is a U.S. and European EPG RIM 412. The U.S. and European EPG RIM 412 is then itself adjusted in an adjustment step 414 utilizing global EPG data specifications 416. The result of the adjustment step 414 is a global EPG RIM 418. This global EPG RIM 418 is subsequently updated in a refinement step 420 based on updated EPG data specifications 422.

In accordance with yet another aspect of the invention, a process is provided for transforming existing content-related information for use with other EPGs.

Fig. 5 is a flow diagram of an illustrative embodiment of the above-noted process. This embodiment of the process utilizes the RIM 200 to transform a set of XML documents generated for use with an "external" EPG so as to be suitable for processing on another, different type of EPG. The FIG. 5 process includes steps 501, 502, 503, 504 and 505, also denoted generally as steps 1 through 5, respectively.

In step 501, working XML schema are generated or otherwise obtained from a set of external XML EPG documents 510. The result of step 501 is one or more XML schema 512 for the external EPG.

An EPG RIM 514 is used in the FIG. 5 process. It is assumed without limitation that the EPG RIM 514 is associated with one or more particular EPG developers, e.g., Philips Electronics (hereinafter "Philips"). The documents 510 are referred to as "external" in this embodiment in that they are assumed to be associated with another EPG developer external to the particular EPG developer(s).

In step 502, the XML schema 512 for the external EPG are related to the EPG RIM 514, e.g., by checking for syntax and semantics consistency. Then, in step 503, one or more Philips XML schema 516 are generated from an appropriate subset of the EPG RIM 514 as identified in step 502.

Step 504 utilizes the external XML schema 512 and the Philips XML schema 516 to generate XML patterns and XSLt transformation rules. The result of this step is an XML stylesheet 518 for transforming the external XML EPG documents 510 to new XML documents 520 that are compliant with the Philips EPG RIM 514. The XML stylesheet 518 may be applied to a conventional XSLt transformation engine as indicated in step 505, so as to transform the external XML EPG documents to the new Philips RIM-compliant XML EPG documents 520.

The FIG. 5 process thus maps the external EPG XML schema 512 to an appropriate subset of the EPG RIM 514 so as to create new, RIM-compliant schema 516. The RIM-compliant schema 516 are further processed to generate the XSL stylesheet 518 that is utilized to convert the external EPG XML documents 510 to the RIM-compliant documents 520.

The use of a RIM such as that described above for generation of EPG XML documents provides a number of significant advantages relative to conventional techniques. For example, it allows XML documents developed for use with particular types of EPGs to be utilizable in an efficient manner on other EPGs. In addition, this approach eases semantic verification of EPG XML conformance claims between applications from different sources. An appropriate level of conformance can also improve the marketability of a given EPG-based processing device. Moreover, the use of the RIM in the manner described can ensure internal consistency across the product lines of a given device manufacturer.

The invention thus facilitates the processing of EPG XML documents and other content-related information so as to deliver consistent results in an efficient manner.

Fig. 6 shows an example of a processing device 600 in which an EPG configured using the techniques of the invention may be implemented.

The device 600 includes a processor 602 and a memory 604 which communicate over at least a portion of a set 605 of one or more system buses. Also utilizing at least a portion of the set 605 of system buses are a display 606 and one or more input/output (I/O) devices 608.

The device 600 may represent one or more of the devices 110 of the FIG. 1 processing system, or any other type of processing device that incorporates an EPG, and the elements of the device 600 may be conventional elements of such devices.

For example, the processor 602 may represent a microprocessor, central processing unit (CPU), digital signal processor (DSP), or application-specific integrated circuit (ASIC), as well as portions or combinations of these and other processing devices. The memory 604 is typically an electronic memory, but may comprise other types of storage devices, such as disk-based optical or magnetic memory.

The EPG techniques described herein may be implemented in whole or in part using software stored and executed using the respective memory and processor elements of the device 600. For example, one or more of the EPGs 112 of the FIG. 1 system may be implemented at least in part using one or more software programs stored in memory 604 and executed by processor 602. The particular manner in which such software programs may be stored and executed in device elements such as memory 604 and processor 602 is well understood in the art and therefore not described in further detail herein.

It should be noted that the device 600 may include other elements not shown, or other types and arrangements of elements capable of providing the EPG processing functions described herein.

The Fig. 6 processing device may also be used to process content-related information for delivery to another processing device equipped with an EPG, e.g., using the techniques described in conjunction with Figs. 3, 4 and 5.

The above-described embodiments of the invention are intended to be illustrative only. For example, the invention can be used in other types of information processing systems and devices using other arrangements of processing elements. In addition, as indicated above, the particular details of the RIM used in a given embodiment of the invention will vary depending upon the type of content to be delivered. These and numerous other alternative embodiments within the scope of the following claims will be apparent to those skilled in the art.

Appendix 1

Class name:
cast

Category:	Logical View
External Documents:	
Export Control:	Public
Cardinality:	n
Hierarchy:	
Superclasses:	personnel
State machine:	No
Concurrency:	Sequential
Persistence:	Transient

Class name:
credits

Category:	Logical View
External Documents:	
Export Control:	Public
Cardinality:	n
Hierarchy:	
Superclasses:	personnel
State machine:	No
Concurrency:	Sequential
Persistence:	Transient

Class name:
DesignatedMarketArea

Category:	Logical View

External Documents:
Export Control: Public
Cardinality: n
Hierarchy:
Superclasses: none
Associations:

<no rolename> : Station in association <unnamed>

Implementation:
Attributes:
name : String
num : unsigned int
Rank : unsigned int

State machine: No
Concurrency: Sequential
Persistence: Transient

Class name:
StationTimeZone

Category: Logical View
Stereotype: enumeration
External Documents:
Export Control: Public
Cardinality: n
Hierarchy:
Superclasses: none
Associations:

<no rolename> : Station in association <unnamed>

Implementation:

Attributes:

GMT
Newfoundland_D_S
Newfoundland
Atlantic_D_S
Atlantic
Eastern_D_S
Eastern
Central_D_S
Central
Mountain_D_S
Mountain
Pacific_D_S
Pacific
Hawaiian_D_S
Hawaiian

State machine: No
Concurrency: Sequential
Persistence: Transient

Class name:
Station

Category: Logical View
External Documents:
Export Control: Public
Cardinality: n
Hierarchy:
Superclasses: none
Associations:

<no rolename> : DesignatedMarketArea in association <unnamed>

<no rolename> : StationTimeZone in association <unnamed>

<no rolename> : Program in association <unnamed>

Implementation:

Attributes:

uid: unsigned int
Unique station ID number.

name : String
Long name of the station.

call_sign : String
Mnemonic or FCC-recognized call sign for long name of the station.
Synonym: call_letters

affiliate : String
Network, cable or broadcasting group the station is associated with. Also it could be a code from the Network Syndication Source.

fcc_channel_number : unsigned int
FCC channel number of the broadcast station.

city : String
City where station is located.

state : String
State where station is located.

zip_code : unsigned int
Postal zip code where station is located.

country : String
Country where station is located.

time_zone : StationTimeZone
Native time zone of the station.

State machine: No
Concurrency: Sequential
Persistence: Transient

Class name:
LiveStatus

Category: Logical View
Stereotype: enumeration
External Documents:
Export Control: Public
Cardinality: n
Hierarchy:
Superclasses: none
Associations:

<no rolename> : Program in association <unnamed>
Implementation:
Attributes:
live
tape
delay

State machine: No
Concurrency: Sequential
Persistence: Transient

Class name:
ProgramLanguage

Category: Logical View
Stereotype: enumeration
External Documents:
Export Control: Public
Cardinality: n
Hierarchy:
Superclasses: none
Associations:

<no rolename> : Program in association <unnamed>

Implementation:
Attributes:
English
French
German
Italian
Portuguese
Spanish

State machine: No
Concurrency: Sequential
Persistence: Transient

Class name:
ColorCode

Category: Logical View
Stereotype: enumeration
External Documents:
Export Control: Public
Cardinality: n
Hierarchy:
Superclasses: none
Associations:

<no rolename> : Program in association <unnamed>

Implementation:
Attributes:
BlackAndWhite
Color
ColorAndBlackAndWhite
Colorized

State machine: No
Concurrency: Sequential
Persistence: Transient

Class name:
PremiereFinal

Category: Logical View
Stereotype: enumeration
External Documents:
Export Control: Public
Cardinality: n

Hierarchy:
Superclasses: none
Associations:
<no rolename> : Program in association <unnamed>
Implementation:
Attributes:
Premiere
Season Premiere
Series Premiere
Season Fianle
Series Finale

State machine: No
Concurrency: Sequential
Persistence: Transient

Class name:
ShowType

Category: Logical View
Stereotype: enumeration
External Documents:
Export Control: Public
Cardinality: n
Hierarchy:
Superclasses: none
Associations:
<no rolename> : Program in association <unnamed>
Implementation:
Attributes:
PaidProgramming
Serial
Series

LimitedSeries
ShortFilm
MiniSeries
Special

State machine: No
Concurrency: Sequential
Persistence: Transient

Class name:
Holiday

Category: Logical View
Stereotype: enumeration
External Documents:
Export Control: Public
Cardinality: n
Hierarchy:
Superclasses: none
Associations:
<no rolename> : Program in association <unnamed>
Implementation:
Attributes:
Christmas
Easter
Thanksgiving
FourthOfJuly
YomKippur
LaborDay
NewYearsEve
NewYearsDay
Hanukkah
ValentineDay

Halloween
SaintPatricksDay

State machine: No
Concurrency: Sequential
Persistence: Transient

Class name:
NetworkSyndicationSource

Category: Logical View
Stereotype: enumeration
External Documents:
Export Control: Public
Cardinality: n
Hierarchy:
Superclasses: none
Associations:
<no rolename> : Program in association <unnamed>
Implementation:
Attributes:
abc
cbs
nbc
pbs
wb
upn
synd
syn89
hbc
si
src
tqs

tva
cbc
ctv
uni
talk

State machine: No
Concurrency: Sequential
Persistence: Transient

Class name:
NetworkSyndicationType

Category: Logical View
Stereotype: enumeration
External Documents:
Export Control: Public
Cardinality: n
Hierarchy:
Superclasses: none
Associations:
<no rolename> : Program in association <unnamed>
Implementation:
Attributes:
BroadcastNetwork
FirstRunSyndication
OffNetwork
CashBarter

State machine: No
Concurrency: Sequential
Persistence: Transient
Class name:

SourceType

Category: Logical View
Stereotype: enumeration
External Documents:
Export Control: Public
Cardinality: n
Hierarchy:
Superclasses: none
Associations:

<no rolename> : Program in association <unnamed>

Implementation:
Attributes:
Block
Local
Network
Syndicated

State machine: No
Concurrency: Sequential
Persistence: Transient

Class name:
AdvisoryDescription

Category: Logical View
Stereotype: enumeration
External Documents:
Export Control: Public
Cardinality: n
Hierarchy:

Superclasses: none

Associations:

<no rolename> : Program in association <unnamed>

Implementation:

Attributes:

AdultSituations
AdolescentesYAdultos
Adultos
BriefNudity
GraphicLanguage
GraphicViolence
Language
MildViolance
Nudity
PublicoGeneral
Rape
StrongSexualContent
Violence

State machine: No
Concurrency: Sequential
Persistence: Transient

Class name:

Program

Category: Logical View

Documentation:

Rating to indicate

External Documents:

Export Control: Public

Cardinality: n

Hierarchy:

Superclasses: none

Associations:

<no rolename> : Station in association <unnamed>

<no rolename> : LiveStatus in association <unnamed>

<no rolename> : ProgramLanguage in association <unnamed>

<no rolename> : ColorCode in association <unnamed>

<no rolename> : PremiereFinal in association <unnamed>

<no rolename> : ShowType in association <unnamed>

<no rolename> : Holiday in association <unnamed>

<no rolename> : NetworkSyndicationSource in association <unnamed>

<no rolename> : NetworkSyndicationType in association <unnamed>

<no rolename> : SourceType in association <unnamed>

<no rolename> : AdvisoryDescription in association <unnamed>

<no rolename> : ProgramRating in association <unnamed>

Implementation:

Attributes:

uid : String

Uniqe Station ID Number.

title : String

Official name by which a movie, show, episode or sports event is known.

alternate_title : String

Alias name for program title; for example the title "Paid

program" is stored here.

reduced_titles : String [ ]

They are reduced program titles.

They are generally used for grid fitting. Each reduced title is shorter than the previous, depending on the length of the actual title and the duration of the program. For instance, a two-hour program with a long title may have 4 alternate titles, and another two-hour program with a very short title, may have no reduced title. The shortest alternate title is always edited to nine characters and the next shortest is edited to thirteen charcters.

subtitle : String

The subtitle field has the same specifications as the title field. In the case of sports, this field will contain the sports team name(s). For example,

Title = Major League Baseball;

Subtitle = Chicago Cubs vs. St.

Louis Cardinals.

The subtitle is used when a program is commonly known by its umbrella title. For example, Masterpiece Theater will always have a subtitle designating the series currently being played. The main factor taken into consideration is whether the show has a recognition factor with the umbrella title or the umbrella title tells the reader more about the show. Anthology shows such as "Wonderful World of Disney" and "Masterpiece Theater" do use the subtitle field.

Team vs. Team, Playoff Sport,

Sporting Event, Sports Related and Sports Anthology use the subtitle field. Team names,

event names, the school from coach's shows, etc. go in the subtitle field. Pseudo-Sports never use the subtitle field.

reduced_subtitles : String [ ]

They are reduced program subtitles. They are generally used for grid fitting. Each reduced subtitle is shorter than the previous, depending on the length of the actual title and the duration of the program.

description : String

String that describes the show, episode, or movie content and includes embedded actors within this description.

The description fields are used to describe the action taking place in the program. We used all three fields in some cases, two fields in others and only one in some, generally broken down by program type.

Most programs must have a first description. The description length is generally determined by the duration of the program.
A 30-minute program's description should not exceed 18 words.
A 60-minute program's should not exceed 25 words.
Longer programs can be written according to their content, for example, the Academy Awards is 3-hours and 30-minutes. The description can be longer than 25 words but should not be too long. Judgment is called for.

alternate_description : String

Alternate Description is used primarily by the Network and Movie editors. The network editors copy the first description and embed the actors' names into the description. The movie editors write a different movie description using the length guidelines of the first description.

reduced_descriptions : String [ ]

Reduced descriptions are also known as the grid descriptions.
This field is regulated by length. Descriptions cannot exceed 45 characters per half-hour (including the title and subtitle fields). Style and language may be forsaken for fit but it should always create a readable, logical sentence. It is not necessary to fit the 45/90

parameters. This description is necessary for any show that falls between 6PM and midnight local time.

Reduced description is also used for 22/44 descriptions on the Talk Show program type. This includes any talk show that falls outside of the time parameters for 45/90. Examples: Good Morning America, Late Night Show, ... etc.

genre_description : String

Description of words or group of words that generally describe a show, episode, movie or sports event.

advisory_descriptions : AdvisoryDescription [ ]

Enumeration of the notation.

air_date : unsigned int

Date the program airs. The date will change from one date to the next at a start of the client-specified day cycle. For example, if the day cycle begins at 2:00 AM, the date will change at 2:00 AM. The day cycle may begin any time, but may not exceed 24 hours. The default cycle starts at 12:00AM.

air_time unsigned int

Time of day the program airs;
hhmm military format.
Synonym: start_time.

duration : unsigned int

Calculated by subtracting the current program's air time from the subsequent program's air time; hhmm format.

part_num : unsigned int

When a program is split into 2 or more viewings, this designates which one it is.

num_of_parts : unsigned int

Designates when a program is split into 2 or more parts for viewing.

repeat : bool = false

designator for a program which has aired previously.

network_syndication_source : NetworkSyndicationSource

Network the program originates from.

network_syndication_type : NetworkSyndicationType

To specify broadcast network, first run syndicated, cash barter and off network programming.

enhanced : bool

Designates enhances program information.

program_language : ProgramLanguage

Language of the copy (description) of the program.

source_type : SourceType

To specify network, local, syndicated and multiple block programming.

show_type : ShowType

Designates series, special, miniseries, limited series, paid programming ...

holiday : Holiday

Description of recognized or traditional holiday.

subtitled : bool = false

Used for foreign movies and shows, if the audio is in a foreign language, the English translation is printed on-screen.

premiere_final : PremiereFinal

Designates a program's premiere or final episode, if applicable.

cable_in_the_classroom : bool = false

Designates the show is available through the Cable in the Classroom program.

secondary_audio_program : bool = false

Designates if the program is subject to Secondary Audio Programming coding.

live_tape_delay : LiveStatus

Designate whether a sports event is being played live, from videotape or a delayed feed.

joined_in_progress : bool = false

Joined in progress when a station begins airing a program at a time other than the official start time.

blackout : bool = false

Designates if the program is subject to blackout restrictions.

hdtv : bool = true

Designates if a show is broadcast in HDTV.

closed_captioning : bool = false

Close Captioning: spoken content of program listed on-screen for the hearingimpaired.

stereo : bool = true

Value that designates if a show, episode, movie or sports event is being broadcast in stereo.

three_d : bool

Designates show in 3-D.

letterbox : bool

Designates program is a letterbox version.

color_code : ColorCode

Designates if a program was produced in color or back/white.

rating : ProgramRating

State machine: No
Concurrency: Sequential
Persistence: Transient

Class name:
TVRating

Category: Logical View
Stereotype: enumeration
External Documents:
Export Control: Public
Cardinality: n
Hierarchy:
Superclasses: none
Associations:

<no rolename> : ProgramRating in association <unnamed>

Implementation:
Attributes:

TVY
TV7

TVG
TVPG
TV14
TVM

State machine: No
Concurrency: Sequential
Persistence: Transient

Class name:
ProgramRating

Category: Logical View
External Documents:
Export Control: Public
Cardinality: n
Hierarchy:
Superclasses: none
Associations:

<no rolename> : Program in association <unnamed>
<no rolename> : TVRating in association <unnamed>

Implementation:
Attributes:
tv_rating : TVRating
TV Parental Guidelines in text form.

dialog_rating : bool
Rating to indicate strong dialog.

fantasyviolance_rating : bool
Rating to indicate fantasy violance.

language_rating : bool
Rating to indicate strong language.

sex_rating : bool
Rating to indicate adult situations.

violence_rating : bool
Rating to indicate strong violence.

adult_language : bool
adult_situations : bool
brief_nudity : bool
graphic_violence : bool
mild_violence : bool
nudity : bool
strong_sexual_content : bool
violence : bool

State machine: No
Concurrency: Sequential
Persistence: Transient

Class name:
RoleDescription

Category: Logical View
Stereotype: enumeration
External Documents:
Export Control: Public
Cardinality: n
Hierarchy:
Superclasses: none

Associations:

<no rolename> : personnel in association <unnamed>

Implementation:

Attributes:

Actor
GuestStar
Director
ExecutiveProducer
Host
Producer
Writer

State machine: No
Concurrency: Sequential
Persistence: Transient

Class name:

Episode

Category: Logical View
External Documents:
Export Control: Public
Cardinality: n
Hierarchy:
Superclasses: Program
Associations:

<no rolename> : personnel in association <unnamed>

Implementation:

Attributes:

syndicated_number : unsigned int

Distributor-designated number corresponding to an episode associated with a specific show.

alt_syndicated_number : unsigned int

Alternate numbering system for syndicated programming. Can differ from syndicated numbering system.

episode_title : String

Descriptive title within the episode. This field is used only for series that designate a specific name for each episode. All wording must be used including "The," "A" and "An". All punctuation must be used except the closing period.

State machine: No
Concurrency: Sequential
Persistence: Transient

Class name:
personnel

Category: Logical View
External Documents:
Export Control: Public
Cardinality: n
Hierarchy:
Superclasses: none
Associations:

<no rolename> : RoleDescription in association <unnamed>
<no rolename> : Episode in association <unnamed>
<no rolename> : movie in association <unnamed>

Implementation:

Attributes:

first_name : String

First name.

last_name : String

Last name.

role_description : RoleDescription

A role for an actor in a movie, or a role describes the program credits (director, executive producer, or a host of a movie or a show).

State machine: No

Concurrency: Sequential

Persistence: Transient

Class name:

MPAARating

Category: Logical View

Documentation:

Only unaltered movies as the MPAA (Motion Picture Association of America) reviewed them can be given the rating. Any alternation, such as changing the language or editing the film or inserting commercials, will void the MPAA rating

Stereotype: enumeration

External Documents:

Export Control: Public

Cardinality: n

Hierarchy:

Superclasses: none

Associations:

<no rolename> : movie in association <unnamed>

Implementation:

Attributes:

AO

G

General

PG

Parental Guidance recommended.

PG-13

Parental Guidance for children under 13.

R

Restricted.

NC-17

No Children under 17.

NR

MA

Mature Audiences.

X

X-rated.

State machine: No
Concurrency: Sequential
Persistence: Transient

Class name:
movie

Category: Logical View
Documentation:
These ratings are researched and given tot the movie by the movie editors.

External Documents:
Export Control: Public
Cardinality: n
Hierarchy:
Superclasses: Program
Associations:

<no rolename> : personnel in association <unnamed>
<no rolename> : MPAARating in association <unnamed>
<no rolename> : StarRating in association <unnamed>

Implementation:
Attributes:
mpaa_rating : MPAARating
Rating supplied by the Motion Picture Association of America.

star_rating : StarRating
An arbitrary critical rating from 1/2 to 4 stars.

run_time : unsigned int
Actual duration which air on pay cable services such as HBO,

CINMAX, ... etc.

country_of_origin : String

Used to distinguished between domestic and foreign films.

made_for_tv : bool = true

Designator of films that was made specifically for broadcast on TV.

release_year : unsigned int

The year in which a movie was released.

production_year : unsigned int

The year in which a movie was produced.

State machine: No
Concurrency: Sequential
Persistence: Transient

Class name:
StarRating

Category: Logical View
Stereotype: enumeration
External Documents:
Export Control: Public
Cardinality: n
Hierarchy:
Superclasses: none
Associations:

<no rolename> : movie in association <unnamed>

Implementation:

Attributes:

One
OnePlus
Two
TwoPlus
Three
ThreePlus
Four

State machine: No
Concurrency: Sequential
Persistence: Transient

Association:

Derived: No
Direction: <non-directional>
Association Class: none

Role:

Class: DesignatedMarketArea
Cardinality / Multiplicity: 1
Navigable: Yes
Aggregate: No
Static: No
Friend: No
Access: Public
Containment: Unspecified

Role:

Class: Station
Cardinality / Multiplicity: 1..n
Navigable: Yes
Aggregate: No
Static: No
Friend: No
Access: Public
Containment: Unspecified

Association:

Derived: No
Direction: <non-directional>
Association Class: none

Role:

Class: Station
Cardinality / Multiplicity:
Navigable: Yes
Aggregate: No
Static: No
Friend: No
Access: Public
Containment: Unspecified

Role:

Class: StationTimeZone
Cardinality / Multiplicity:
Navigable: Yes
Aggregate: No
Static: No
Friend: No

Access: Public
Containment: Unspecified

Association:

Derived: No
Direction: <non-directional>
Association Class: none

Role:
Class: Station
Cardinality / Multiplicity: 1..n
Navigable: Yes
Aggregate: No
Static: No
Friend: No
Access: Public
Containment: Unspecified

Role:
Class: Program
Cardinality / Multiplicity: 1..n
Navigable: Yes
Aggregate: No
Static: No
Friend: No
Access: Public
Containment: Unspecified

Association:

Derived: No

Direction: <non-directional>
Association Class: none

Role:
Class: LiveStatus
Cardinality / Multiplicity:
Navigable: Yes
Aggregate: No
Static: No
Friend: No
Access: Public
Containment: Unspecified

Role:
Class: Program
Cardinality / Multiplicity:
Navigable: Yes
Aggregate: No
Static: No
Friend: No
Access: Public
Containment: Unspecified

Association:

Derived: No
Direction: <non-directional>
Association Class: none

Role:
Class: Program
Cardinality / Multiplicity:
Navigable: Yes

Aggregate: No
Static: No
Friend: No
Access: Public
Containment: Unspecified

Role:
Class: ProgramLanguage
Cardinality / Multiplicity:
Navigable: Yes
Aggregate: No
Static: No
Friend: No
Access: Public
Containment: Unspecified

Association:

Derived: No
Direction: <non-directional>
Association Class: none

Role:
Class: Program
Cardinality / Multiplicity:
Navigable: Yes
Aggregate: No
Static: No
Friend: No
Access: Public
Containment: Unspecified

Role:

Class: ColorCode
Cardinality / Multiplicity:
Navigable: Yes
Aggregate: No
Static: No
Friend: No
Access: Public
Containment: Unspecified

Association:

Derived: No
Direction: <non-directional>
Association Class: none

Role:

Class: Program
Cardinality / Multiplicity:
Navigable: Yes
Aggregate: No
Static: No
Friend: No
Access: Public
Containment: Unspecified

Role:

Class: PremiereFinal
Cardinality / Multiplicity:
Navigable: Yes
Aggregate: No
Static: No
Friend: No
Access: Public

Containment: Unspecified

Association:

Derived: No
Direction: <non-directional>
Association Class: none

Role:
Class: Program
Cardinality / Multiplicity:
Navigable: Yes
Aggregate: No
Static: No
Friend: No
Access: Public
Containment: Unspecified

Role:
Class: ShowType
Cardinality / Multiplicity:
Navigable: Yes
Aggregate: No
Static: No
Friend: No
Access: Public
Containment: Unspecified

Association:

Derived: No
Direction: <non-directional>

Association Class: none

Role:

Class: Program
Cardinality / Multiplicity:
Navigable: Yes
Aggregate: No
Static: No
Friend: No
Access: Public
Containment: Unspecified

Role:

Class: Holiday
Cardinality / Multiplicity:
Navigable: Yes
Aggregate: No
Static: No
Friend: No
Access: Public
Containment: Unspecified

Association:

Derived: No
Direction: <non-directional>
Association Class: none

Role:

Class: Program
Cardinality / Multiplicity:
Navigable: Yes
Aggregate: No

Static: No

Friend: No

Access: Public

Containment: Unspecified

Role:

Class: NetworkSyndicationSource

Cardinality / Multiplicity:

Navigable: Yes

Aggregate: No

Static: No

Friend: No

Access: Public

Containment: Unspecified

Association:

Derived: No

Direction: <non-directional>

Association Class: none

Role:

Class: Program

Cardinality / Multiplicity:

Navigable: Yes

Aggregate: No

Static: No

Friend: No

Access: Public

Containment: Unspecified

Role:

Class: NetworkSyndicationType

Cardinality / Multiplicity:
Navigable: Yes
Aggregate: No
Static: No
Friend: No
Access: Public
Containment: Unspecified

Association:

Derived: No
Direction: <non-directional>
Association Class: none

Role:
Class: SourceType
Cardinality / Multiplicity:
Navigable: Yes
Aggregate: No
Static: No
Friend: No
Access: Public
Containment: Unspecified

Role:
Class: Program
Cardinality / Multiplicity:
Navigable: Yes
Aggregate: No
Static: No
Friend: No
Access: Public
Containment: Unspecified

Association:

Derived: No
Direction: <non-directional>
Association Class: none

Role:
Class: Program
Cardinality / Multiplicity:
Navigable: Yes
Aggregate: No
Static: No
Friend: No
Access: Public
Containment: Unspecified

Role:
Class: AdvisoryDescription
Cardinality / Multiplicity:
Navigable: Yes
Aggregate: No
Static: No
Friend: No
Access: Public
Containment: Unspecified

Association:

Derived: No
Direction: <non-directional>
Association Class: none

Role:

Class: ProgramRating
Cardinality / Multiplicity:
Navigable: Yes
Aggregate: No
Static: No
Friend: No
Access: Public
Containment: Unspecified

Role:

Class: Program
Cardinality / Multiplicity:
Navigable: Yes
Aggregate: No
Static: No
Friend: No
Access: Public
Containment: Unspecified

Association:

Derived: No
Direction: <non-directional>
Association Class: none

Role:

Class: TVRating
Cardinality / Multiplicity:
Navigable: Yes
Aggregate: No

Static: No
Friend: No
Access: Public
Containment: Unspecified

Role:

Class: ProgramRating
Cardinality / Multiplicity:
Navigable: Yes
Aggregate: No
Static: No
Friend: No
Access: Public
Containment: Unspecified

Association:

Derived: No
Direction: <non-directional>
Association Class: none

Role:

Class: RoleDescription
Cardinality / Multiplicity:
Navigable: Yes
Aggregate: No
Static: No
Friend: No
Access: Public
Containment: Unspecified

Role:

Class: personnel

Cardinality / Multiplicity:
Navigable: Yes
Aggregate: No
Static: No
Friend: No
Access: Public
Containment: Unspecified

Association:

Derived: No
Direction: <non-directional>
Association Class: none

Role:

Class: Episode
Cardinality / Multiplicity: 1
Navigable: Yes
Aggregate: Yes
Static: No
Friend: No
Access: Public
Containment: Unspecified

Role:

Class: personnel
Cardinality / Multiplicity: 1..n
Navigable: Yes
Aggregate: No
Static: No
Friend: No
Access: Public
Containment: Unspecified

Association:

Derived: No
Direction: <non-directional>
Association Class: none

Role:
Class: movie
Cardinality / Multiplicity: 1
Navigable: Yes
Aggregate: Yes
Static: No
Friend: No
Access: Public
Containment: Unspecified

Role:
Class: personnel
Cardinality / Multiplicity: 1..n
Navigable: Yes
Aggregate: No
Static: No
Friend: No
Access: Public
Containment: Unspecified

Association:

Derived: No
Direction: <non-directional>
Association Class: none

Role:

Class: movie
Cardinality / Multiplicity:
Navigable: Yes
Aggregate: No
Static: No
Friend: No
Access: Public
Containment: Unspecified

Role:

Class: MPAARating
Cardinality / Multiplicity:
Navigable: Yes
Aggregate: No
Static: No
Friend: No
Access: Public
Containment: Unspecified

Association:

Derived: No
Direction: <non-directional>
Association Class: none

Role:

Class: movie
Cardinality / Multiplicity:
Navigable: Yes
Aggregate: No
Static: No
Friend: No

Access: Public
Containment: Unspecified

Role:
Class: StarRating
Cardinality / Multiplicity:
Navigable: Yes
Aggregate: No
Static: No
Friend: No
Access: Public
Containment: Unspecified

CLAIMS:

1. A method for processing content-related information for delivery to a processing device (110, 600) configured to support an electronic program guide (112) of a first type, the method comprising the steps of:

determining a reference information model (200) for use with the content-related information; and

configuring at least a portion of the content-related information for consistency with corresponding portions of the reference information model, the portion of the content-related information so configured thereby being suitable for processing by at least the electronic program guide of the first type.

2. The method of claim 1 wherein the configured portion of the content-related information is suitable for processing by the electronic program guide of the first type and at least a second electronic program guide of a second type different than the first type.

3. The method of claim 1 wherein the content-related information comprises one or more documents (306, 510, 520) in an extensible mark-up language.

4. The method of claim 1 wherein the reference information model comprises a plurality of classes of information, and specifies properties of the classes utilizing one or more attributes, relationships and states.

5. The method of claim 1 wherein the reference information model comprises a plurality of elements including one or more enumeration elements (202) and one or more class elements (204, 206, 208, 210, 212, 214, 216, 218), a given one of the class elements being associated with at least a subset of the enumeration elements and at least a subset of the remaining class elements.

6. The method of claim 5 wherein the given one of the class elements comprises a program class element, and the remaining class elements comprise one or more of movie, episode, personnel, cast, credits, station and designated market area class elements.

7. The method of claim 5 wherein instances of the classes are configured as objects in an object-oriented programming format, and one or more of the objects contain structures represented as attributes.

8. The method of claim 1 wherein the reference information model is configured in accordance with a unified modeling language format.

9. The method of claim 1 wherein the configuring step further comprises generating one or more schema (304, 512, 516) associated with the electronic program guide of the first type, the schema being generated based at least in part on an associated portion of the reference information model, and utilizing the schema to generate one or more documents comprising the content-related information.

10. The method of claim 9 wherein the configuring step further comprises generating a plurality of different schema, each of the schema being associated with one or more of the electronic program guide of the first type and an electronic program guide of a second type different than the first type, each of the schema being utilized to generate one or more documents comprising the content-related information.

11. The method of claim 1 wherein the reference information model is generated utilizing an iterative process in which an initial version (406) of the model is generated using a first set of data specifications (402), and at least one subsequent version (412, 418) of the model is generated from the initial version using at least a second set of data specifications (410, 416).

12. The method of claim 11 wherein the at least one subsequent version of the model is periodically updated in accordance with one or more sets of updated data specifications (422).

13. The method of claim 1 wherein the configuring step comprises transforming the content-related information from a first format not compliant with the reference information model to a second format compliant with the reference information model.

14. The method of claim 13 wherein the content-related information in the first format comprises one or more documents for use with an electronic program guide of a type not based on the reference information model, and further wherein the documents are converted to the second format so as to be utilizable at least by the electronic program guide of the first type.

15. The method of claim 13 wherein the transforming step utilizes an extensible mark-up language style sheet generated at least in part utilizing the content-related information in the first format and the reference information model.

16. A method for use in a processing device (110, 600) configured to support an electronic program guide (112) of a first type for processing content-related information, the method comprising the steps of:

receiving the content-related information, at least a portion of the content-related information being configured for consistency with corresponding portions of a reference information model (200), the portion of the content-related information so configured thereby being suitable for processing by at least the electronic program guide of the first type; and

processing the content-related information to generate a corresponding output at the processing device.

17. An apparatus for processing content-related information for delivery to a processing device (110, 600) configured to support an electronic program guide (112) of a first type, the apparatus comprising:

a processor (602) operative to configure at least a portion of the content-related information for consistency with corresponding portions of a reference information model (200), the portion of the content-related information so configured thereby being suitable for processing by at least the electronic program guide of the first type; and

a memory (604) coupled to the processor, for at least temporarily storing at least a portion of the content-related information.

18. An apparatus associated with a processing device (110, 600) configured to support an electronic program guide (112) of a first type for processing content-related information, the apparatus comprising:

a processor (602) operative to implement at least a portion of the electronic program guide of the first type for processing the content-related information, at least a portion of the content-related information being configured for consistency with corresponding portions of a reference information model (200), the portion of the content-related information so configured thereby being suitable for processing by at least the electronic program guide of the first type; and

a memory (604) coupled to the processor, for at least temporarily storing at least a portion of the content-related information.

19. A computer program product enabling a programmable device when executing said computer program product to function as the apparatus of claim 17 or 18.

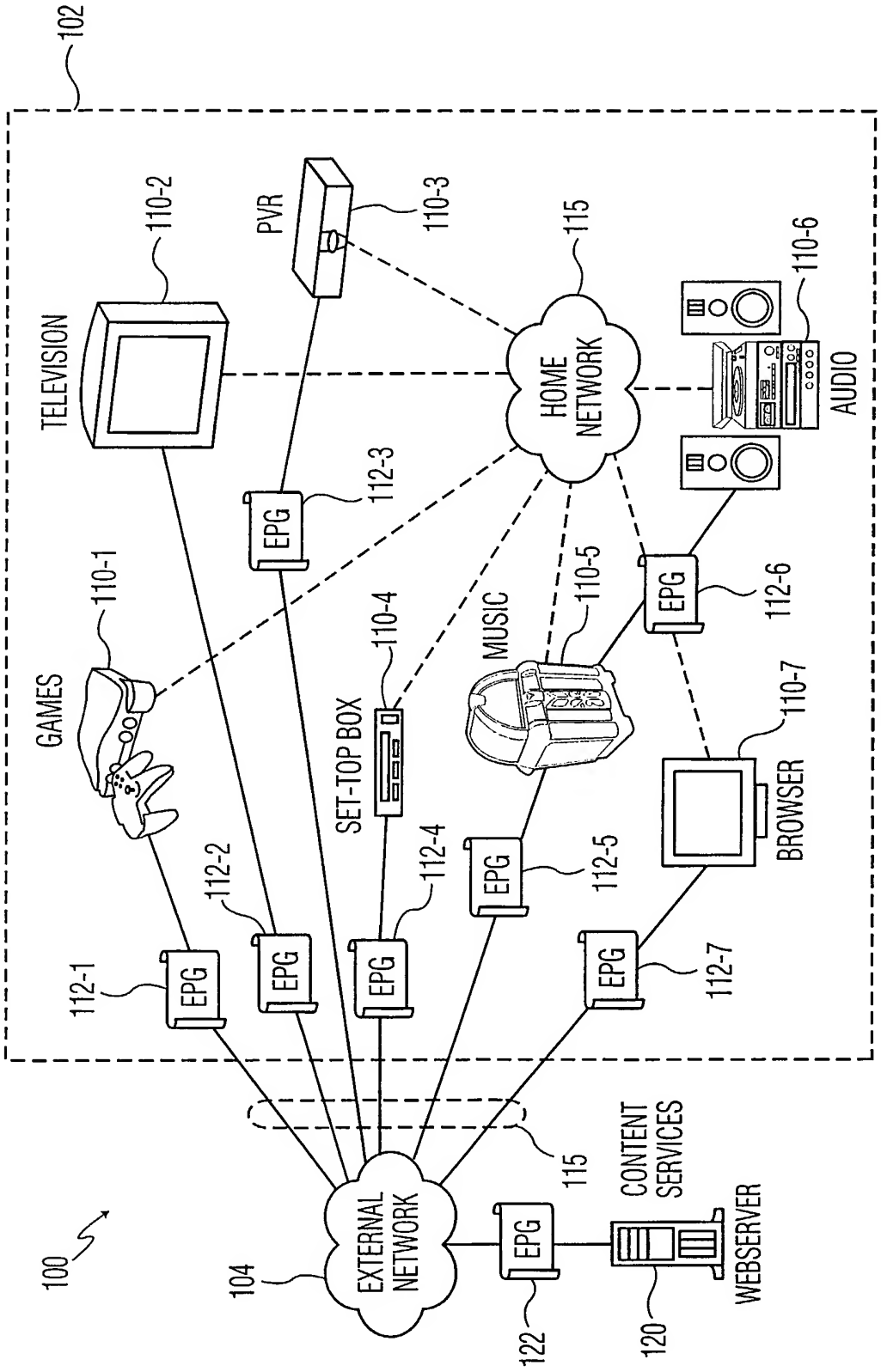

102
TELEVISION
110-2
PVR
110-3
115
HOME NETWORK
110-6
AUDIO
GAMES
110-1
EPG
112-3
MUSIC
110-5
EPG
112-6
SET-TOP BOX
110-4
BROWSER
110-7
EPG
112-4
EPG
112-5
EPG
112-7
EPG
112-2
EPG
112-1
100
104
EXTERNAL NETWORK
115
CONTENT SERVICES
WEBSERVER
EPG
122
120

FIG. 1

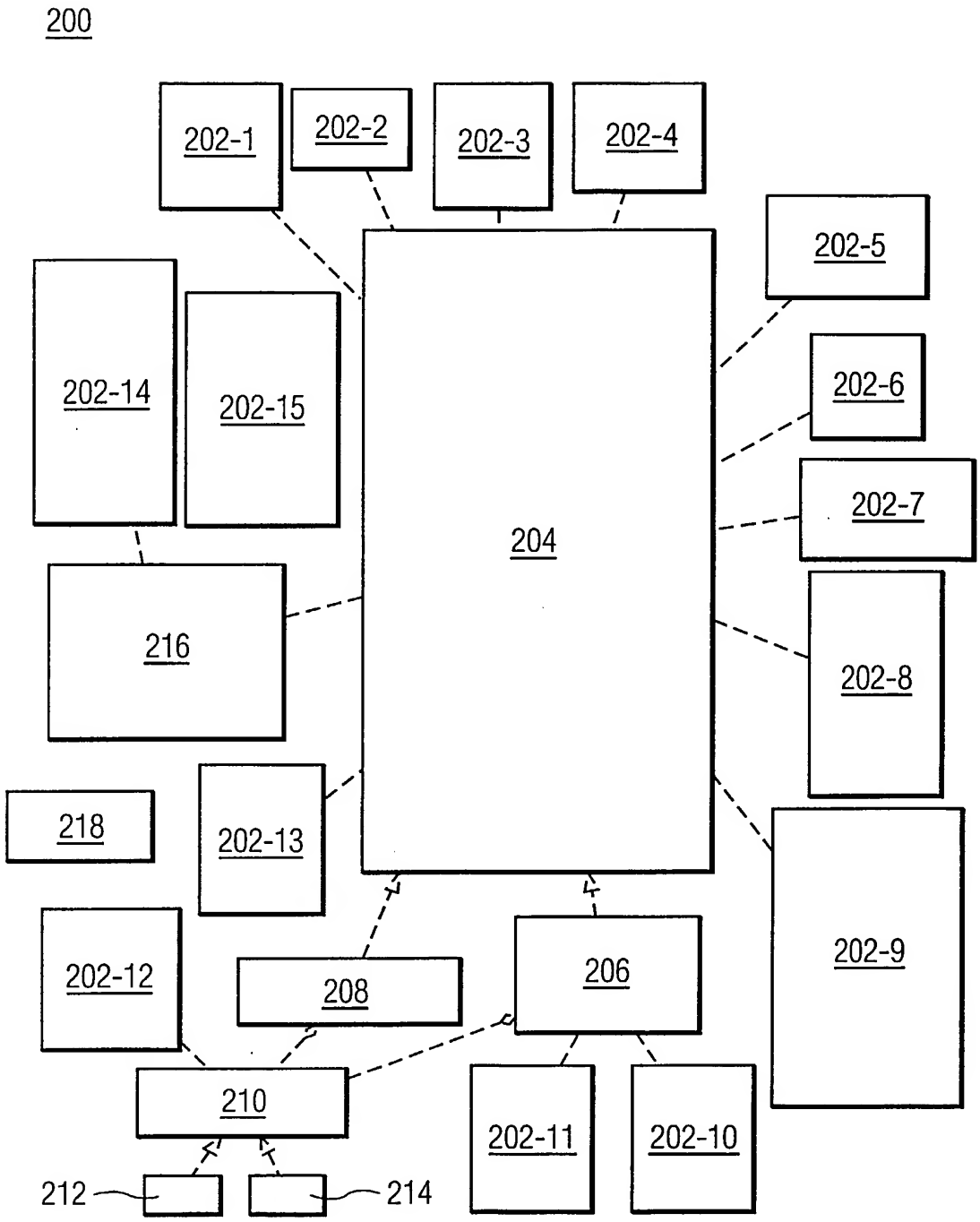

200
202-1
202-2
202-3
202-4
202-5
202-6
202-7
202-8
202-9
202-10
202-11
202-12
202-13
202-14
202-15
204
206
208
210
212
214
216
218

FIG. 2

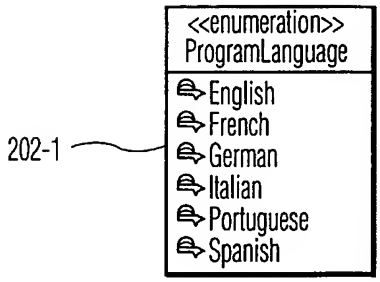


FIG. 2A

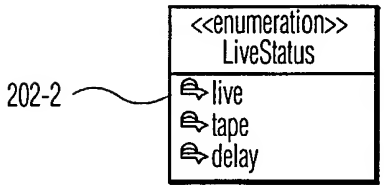


FIG. 2B

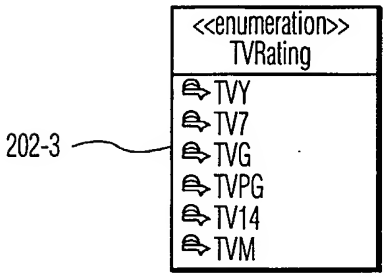


FIG. 2C

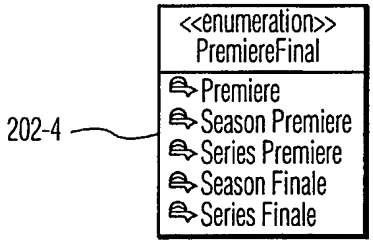


FIG. 2D

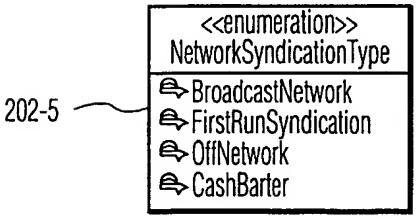


FIG. 2E

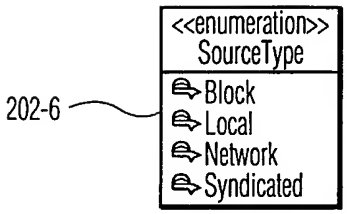


FIG. 2F

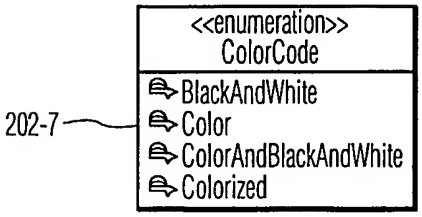


FIG. 2G

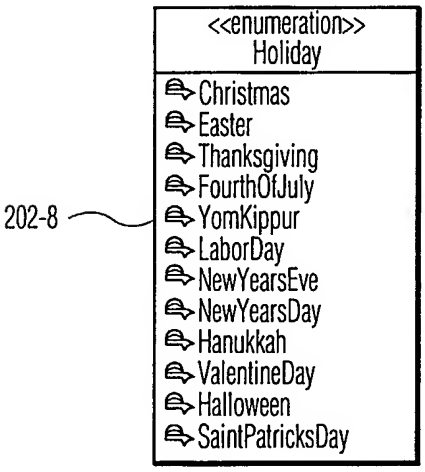


FIG. 2H

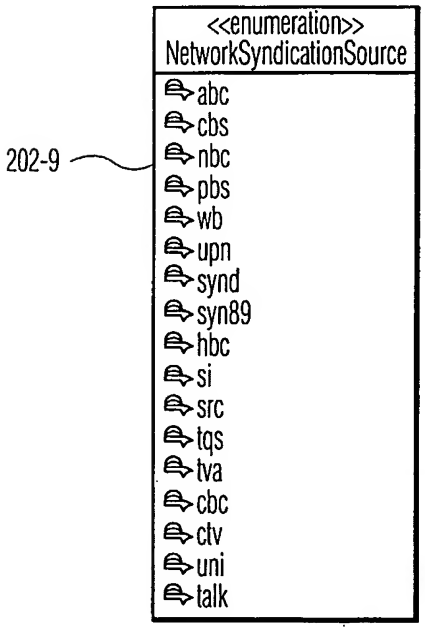


FIG. 2I

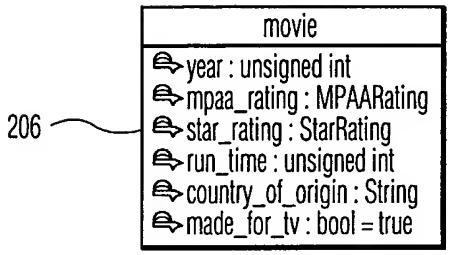


FIG. 2J

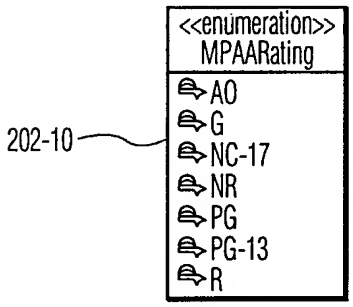


FIG. 2K

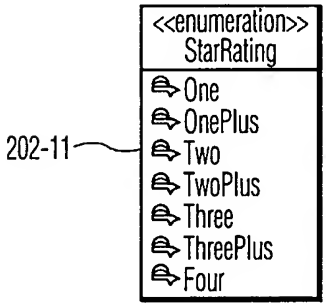


FIG. 2L

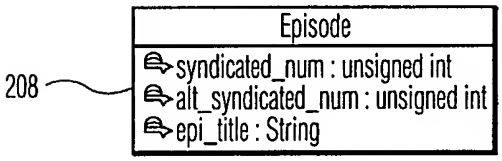


FIG. 2M

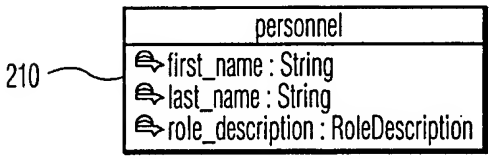


FIG. 2N

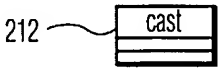


FIG. 2O

FIG. 2P

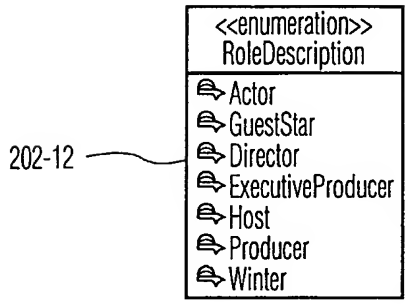


FIG. 2Q

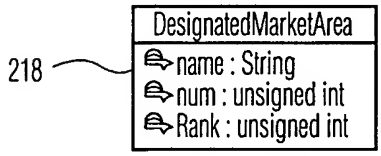


FIG. 2R

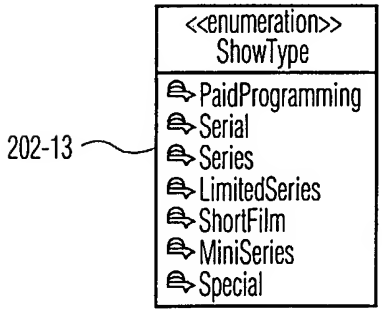


FIG. 2S

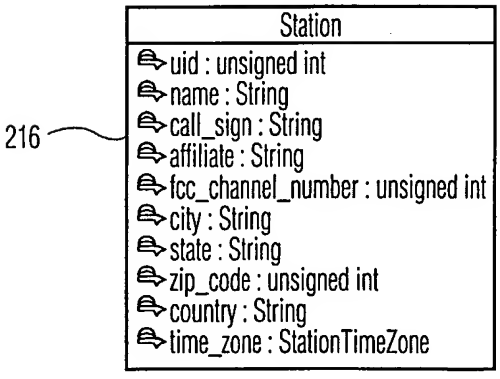


FIG. 2T

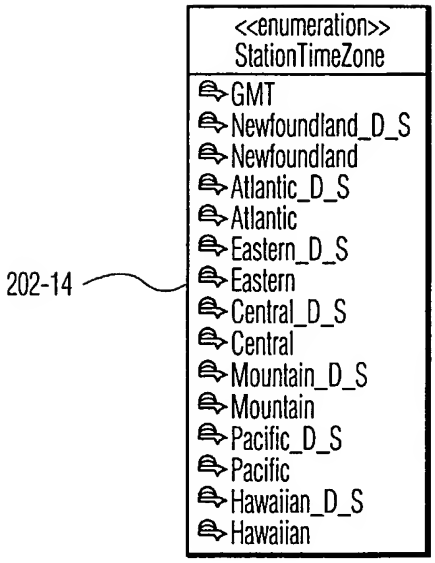

<<enumeration>>
StationTimeZone
GMT
Newfoundland_D_S
Newfoundland
Atlantic_D_S
Atlantic
Eastern_D_S
Eastern
Central_D_S
Central
Mountain_D_S
Mountain
Pacific_D_S
Pacific
Hawaiian_D_S
Hawaiian
202-14

FIG. 2U

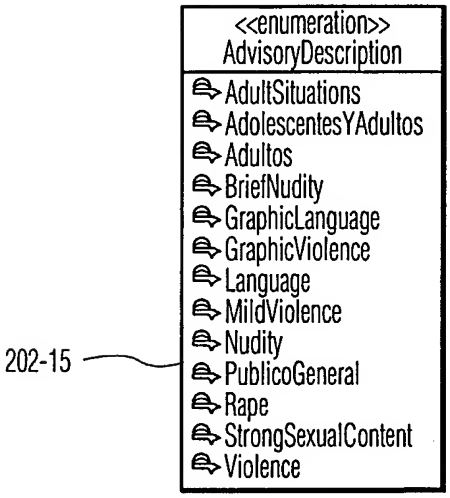

<<enumeration>>
AdvisoryDescription
AdultSituations
AdolescentesYAdultos
Adultos
BriefNudity
GraphicLanguage
GraphicViolence
Language
MildViolence
Nudity
PublicoGeneral
Rape
StrongSexualContent
Violence
202-15

FIG. 2V

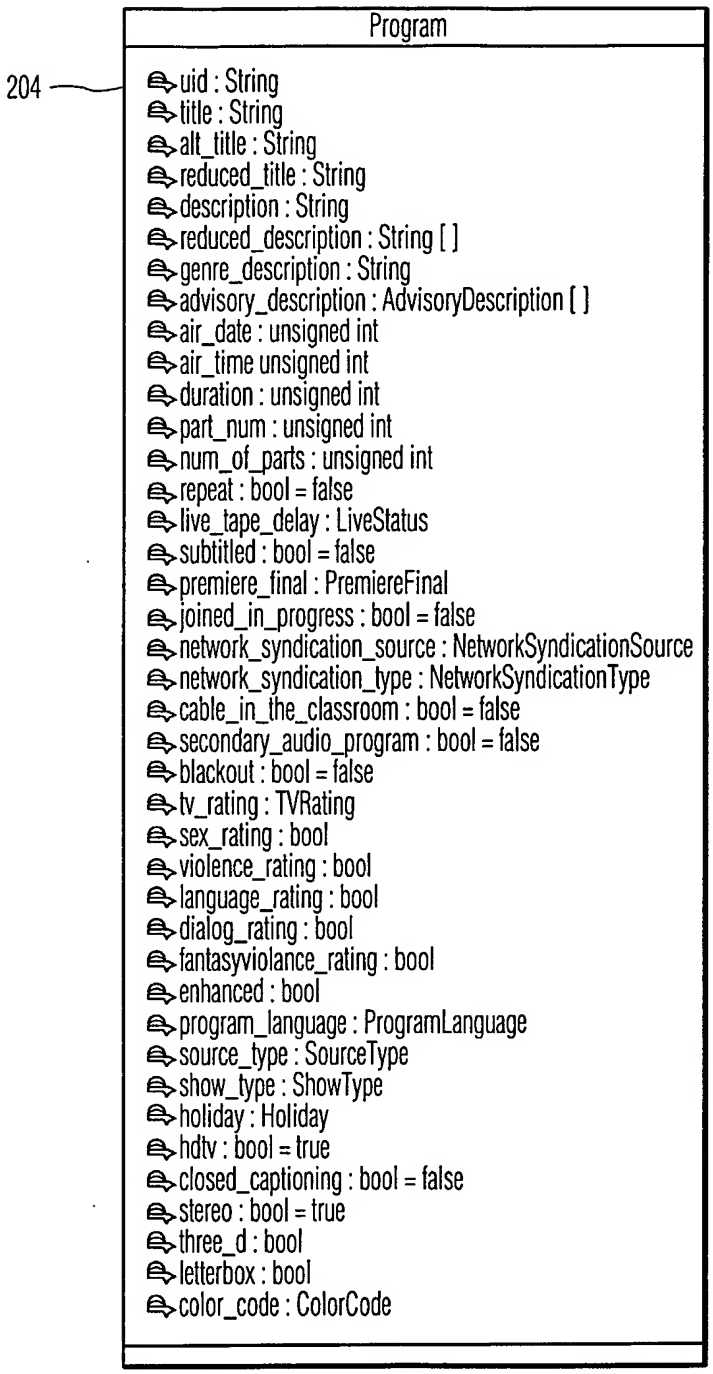

Program
204
uid : String
title : String
alt_title : String
reduced_title : String
description : String
reduced_description : String [ ]
genre_description : String
advisory_description : AdvisoryDescription [ ]
air_date : unsigned int
air_time unsigned int
duration : unsigned int
part_num : unsigned int
num_of_parts : unsigned int
repeat : bool = false
live_tape_delay : LiveStatus
subtitled : bool = false
premiere_final : PremiereFinal
joined_in_progress : bool = false
network_syndication_source : NetworkSyndicationSource
network_syndication_type : NetworkSyndicationType
cable_in_the_classroom : bool = false
secondary_audio_program : bool = false
blackout : bool = false
tv_rating : TVRating
sex_rating : bool
violence_rating : bool
language_rating : bool
dialog_rating : bool
fantasyviolance_rating : bool
enhanced : bool
program_language : ProgramLanguage
source_type : SourceType
show_type : ShowType
holiday : Holiday
hdtv : bool = true
closed_captioning : bool = false
stereo : bool = true
three_d : bool
letterbox : bool
color_code : ColorCode

FIG. 2W

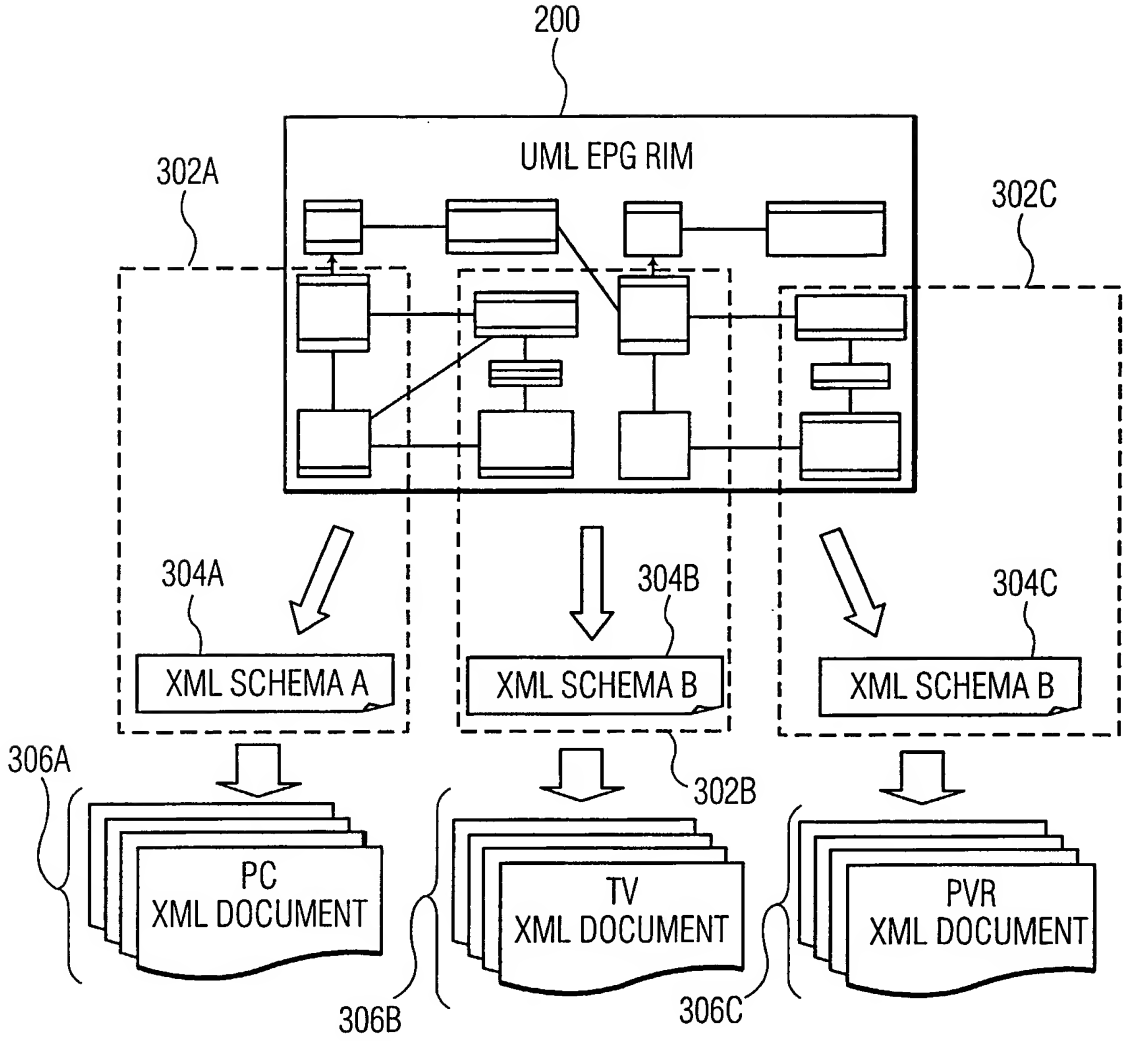

200
UML EPG RIM
302A
302C
304A
304B
304C
XML SCHEMA A
XML SCHEMA B
XML SCHEMA B
306A
302B
PC
XML DOCUMENT
TV
XML DOCUMENT
PVR
XML DOCUMENT
306B
306C

FIG. 3

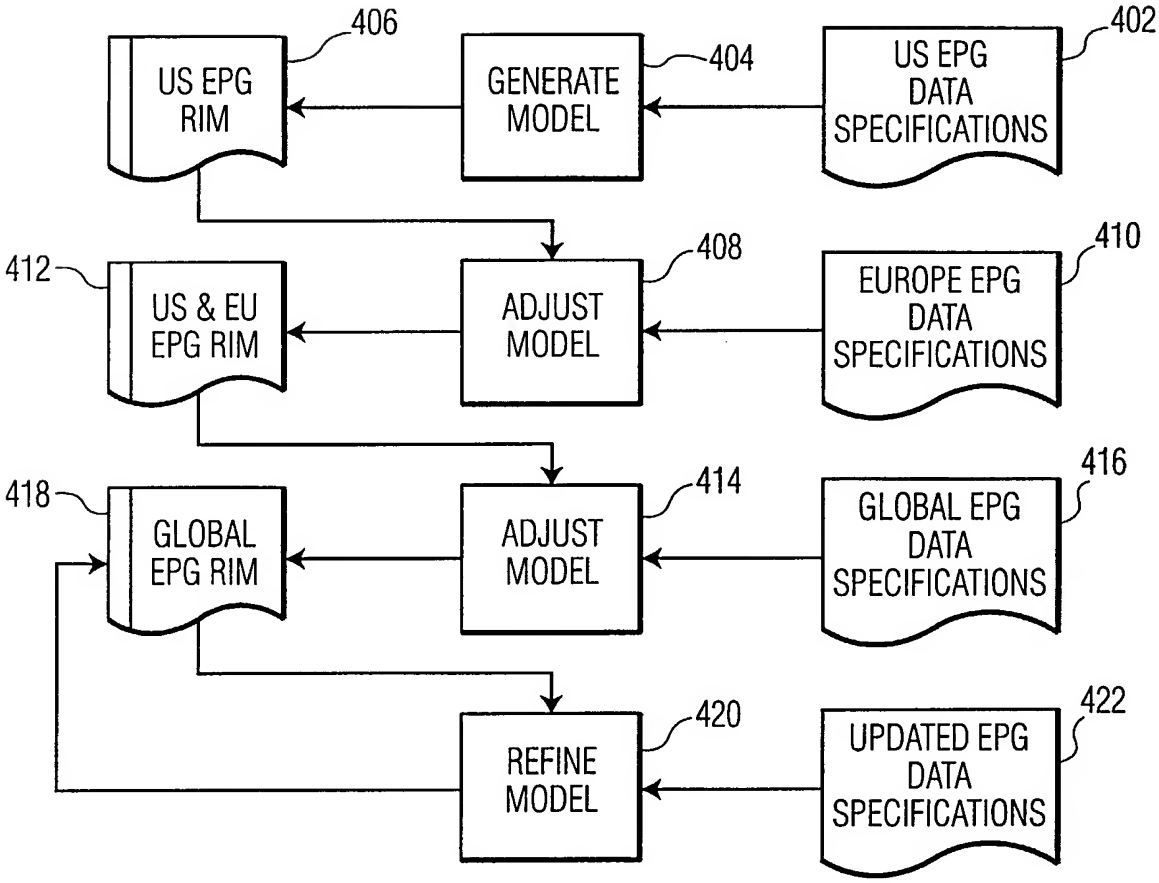

406
US EPG RIM
GENERATE MODEL
404
US EPG DATA SPECIFICATIONS
402
412
US & EU EPG RIM
ADJUST MODEL
408
EUROPE EPG DATA SPECIFICATIONS
410
418
GLOBAL EPG RIM
ADJUST MODEL
414
GLOBAL EPG DATA SPECIFICATIONS
416
REFINE MODEL
420
UPDATED EPG DATA SPECIFICATIONS
422

FIG. 4

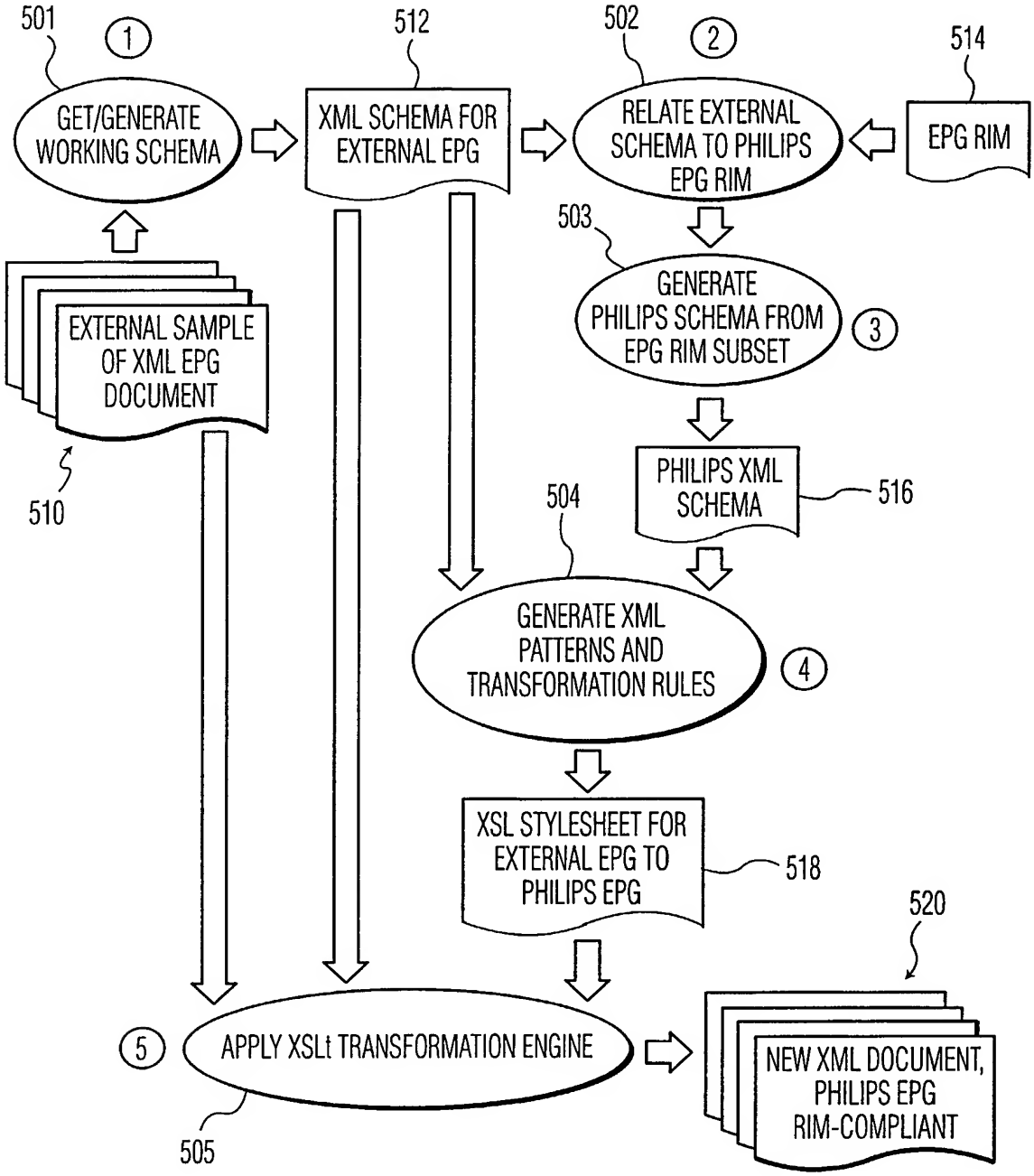

501
1
GET/GENERATE WORKING SCHEMA
512
XML SCHEMA FOR EXTERNAL EPG
502
2
RELATE EXTERNAL SCHEMA TO PHILIPS EPG RIM
514
EPG RIM
503
GENERATE PHILIPS SCHEMA FROM EPG RIM SUBSET
3
EXTERNAL SAMPLE OF XML EPG DOCUMENT
510
PHILIPS XML SCHEMA
516
504
GENERATE XML PATTERNS AND TRANSFORMATION RULES
4
XSL STYLESHEET FOR EXTERNAL EPG TO PHILIPS EPG
518
520
5
APPLY XSLt TRANSFORMATION ENGINE
505
NEW XML DOCUMENT, PHILIPS EPG RIM-COMPLIANT

FIG. 5

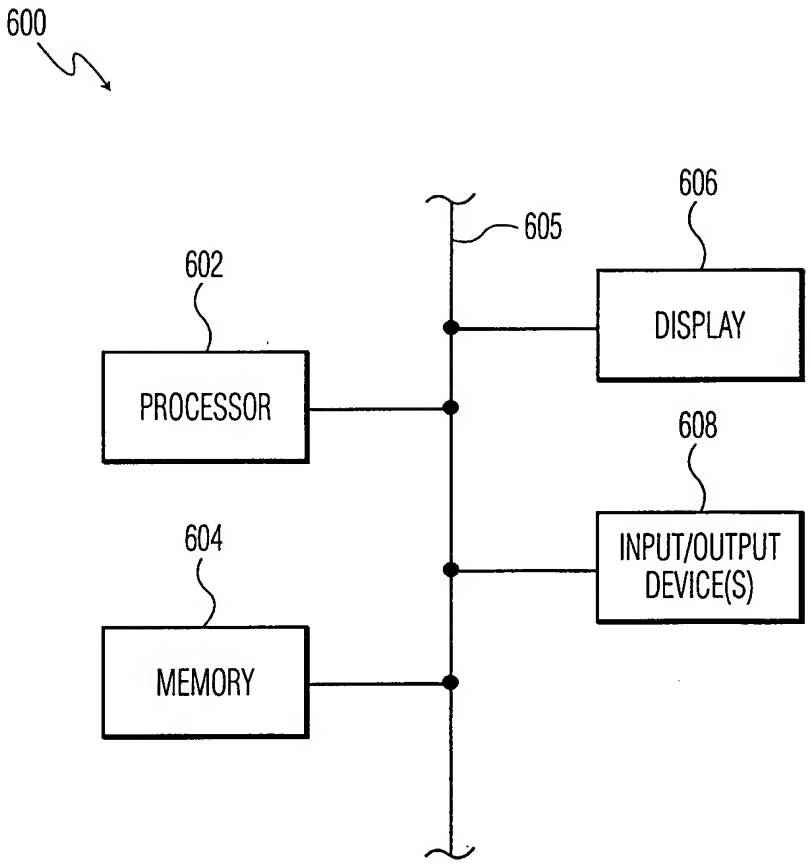

600
606
605
602
DISPLAY
PROCESSOR
608
604
INPUT/OUTPUT
DEVICE(S)
MEMORY

FIG. 6

INTERNATIONAL SEARCH REPORT

Inte nal Application No
PCT/IB 02/02568

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 H04N5/445 G06F17/30

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 H04N G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)
EPO-Internal, WPI Data, PAJ

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 5 675 745 A (OKU MASAYOSHI ET AL) 7 October 1997 (1997-10-07) column 8, line 65 -column 23, line 52	1-19
T	US 2002/122057 A1 (MALONEY KRISELLEN) 5 September 2002 (2002-09-05) paragraph '0028! - paragraph '0105!	1-19

☐ Further documents are listed in the continuation of box C.

☒ Patent family members are listed in annex.

° Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
30 September 2002	07/10/2002

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Materne, A

INTERNATIONAL SEARCH REPORT

nformation on patent family members

Int ınal Application No
PCT/IB 02/02568

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 5675745 A	07-10-1997	JP 2923552 B2	26-07-1999
		JP 8287163 A	01-11-1996
		US 6098047 A	01-08-2000
US 2002122057 A1	05-09-2002	NONE	